

---

UNIP – Universidade Paulista  
Campus Tatuapé – SP  
Ciência da Computação  
Engenharia de Software

Sistemas Legados

Prof. Marcelo Nogueira

---

---

---

---

---

---

---

---

## Sistemas legados

---

- Sistemas de software mais velhos que permanecem vital para uma organização

---

---

---

---

---

---

---

---

## Objetivos

---

- Compreender o que significa o termo 'sistemas legados' e por que esses sistemas são fundamentais para a operação de muitas empresas
- Conhecer as estruturas comuns dos sistemas legados
- Compreender os princípios do projeto orientado a funções – a estratégia de projeto mais comumente utilizada para os atuais sistemas legados
- Compreender como os sistemas legados podem se avaliados para decidir se eles devem ser descartados ou mantidos, se devem passar pelo processo de reengenharia ou ser substituídos

---

---

---

---

---

---

---

---

## Tópicos

---

- Estruturas dos sistemas legados
- Projeto de sistemas legados
- Avaliação dos sistemas legados

---

---

---

---

---

---

---

---

## Sistemas legados

---

- Sistemas de software que são desenvolvidos especialmente para uma organização que tem uma longa vida útil
- Muitos sistemas de software que ainda estão em uso foram desenvolvidos há muitos anos usando tecnologias que já estão obsoletas
- Esses sistemas ainda são críticos de negócios, isto é, eles são essenciais para o funcionamento normal do negócio
- A eles foi dado o nome de sistemas legados

---

---

---

---

---

---

---

---

## Substituição de sistema legado

---

- Há um risco significativo de negócio em simplesmente descartar sistemas legados e substituí-los por um sistema que foi desenvolvido utilizando uma tecnologia moderna
  - Raramente existe uma especificação completa do sistema legado. Se existir uma especificação, é pouco provável que ela incorpore todas as mudanças que foram feitas no sistema
  - Os processos corporativos e o modo como os sistemas legados operam estão sempre intrinsecamente entrelaçados
  - Importantes regras corporativas podem estar inseridas no software e podem não estar documentadas em nenhum outro lugar
  - O desenvolvimento de um software novo é arriscado, uma vez que podem ocorrer problemas inesperados com um novo sistema

---

---

---

---

---

---

---

---

## Alteração de sistemas legados

- O sistema deve mudar para permanecer útil
- No entanto, alterar um sistema legado é muitas vezes dispendioso
  - Diferentes partes do sistema foram implementadas por diferentes equipes. Portanto, não há um estilo de programação consistente
  - O sistema pode utilizar uma linguagem de programação obsoleta
  - Frequentemente, a documentação do sistema é inadequada e desatualizada
  - Em geral, muitos anos de manutenção podem ter corrompido a estrutura do sistema
  - O sistema pode ter sido otimizado para melhorar a utilização de espaço ou a velocidade de execução, em vez de ter sido escrito para facilitar a compreensão
  - Os dados processados pelo sistema podem estar armazenados em diferentes arquivos, que podem ter estruturas incompatíveis

---

---

---

---

---

---

---

---

## Dilema fundamental

- Se continuarem utilizando os sistemas legados e fazendo alterações, seus custos aumentarão
- Se decidirem substituir seus sistemas legados por novos sistemas, isso será dispendioso
- Muitas empresas estão examinando técnicas de engenharia de software que ampliem o tempo de duração dos sistemas legados e que reduzam os custos de manter esses sistemas em uso, como a evolução de produtos de software e a reengenharia de software.

---

---

---

---

---

---

---

---

## Estruturas dos sistema legados

- Os sistemas legados são sistemas sociotécnicos baseados em computadores e não simplesmente sistemas de software antigos
  - Hardware de sistema – como hardware de mainframe
  - Software de apoio – deste sistema operacional e utilitários até compiladores
  - Software de aplicação – fornece os serviços de negócios é composto de vários programas separados, desenvolvidos em diferentes épocas
  - Dados de aplicação – dados processados pelo sistema de aplicação, esses dados podem ser inconsistentes e podem estar duplicados em diferentes arquivos
  - Processos de negócios – os processos utilizados nas empresas a fim de atingir algum objetivo de negócios
  - Políticas e regras de negócios – definições de como a empresa deve ser conduzida e as restrições às quais ela deve se submeter

---

---

---

---

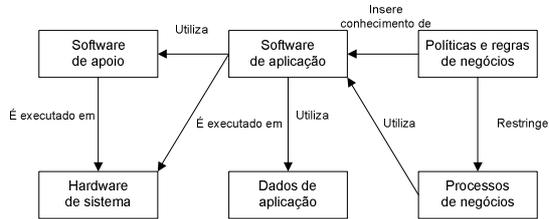
---

---

---

---

## Componentes de sistemas legados



Ofan Sommerville 2000

Engenharia de Software, 6ª edição, Capítulo 26

Slide 10

---

---

---

---

---

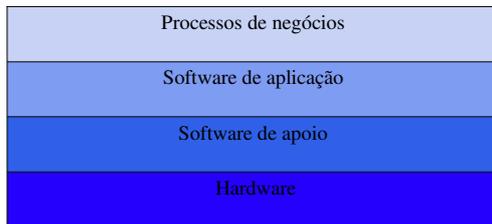
---

---

---

## Modelo em camadas

### Sistema sociotécnico



Ofan Sommerville 2000

Engenharia de Software, 6ª edição, Capítulo 26

Slide 11

---

---

---

---

---

---

---

---

## Modificação de sistema

- Em tese, é possível fazer alterações em uma camada sem afetar qualquer uma das camadas adjacentes
- Na prática, isso raramente funciona, as razões são:
  - A modificação de uma camada pode introduzir novos recursos e as camadas superiores podem ser modificadas para se beneficiarem desses recursos
  - A modificação do software pode torná-lo mais lento, de modo que um novo hardware é necessário, a fim de melhorar o desempenho
  - Muitas vezes, é impossível manter interfaces de hardware, especialmente se for proposta uma mudança radical para um novo tipo de hardware, por exemplo, mudar o hardware de mainframe para sistemas cliente-servidor

Ofan Sommerville 2000

Engenharia de Software, 6ª edição, Capítulo 26

Slide 12

---

---

---

---

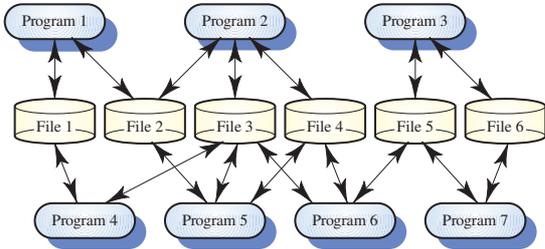
---

---

---

---

## Sistemas legados de aplicação



Ofan Sommerville 2000

Engenharia de Software, 6ª edição, Capítulo 26

Slide 13

---

---

---

---

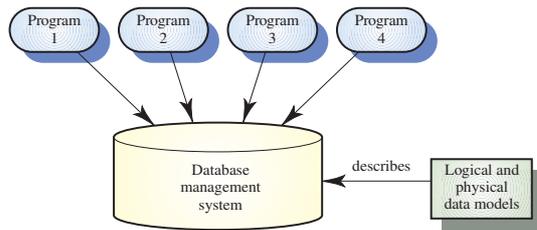
---

---

---

---

## Sistemas centrados em banco de dados



Ofan Sommerville 2000

Engenharia de Software, 6ª edição, Capítulo 26

Slide 14

---

---

---

---

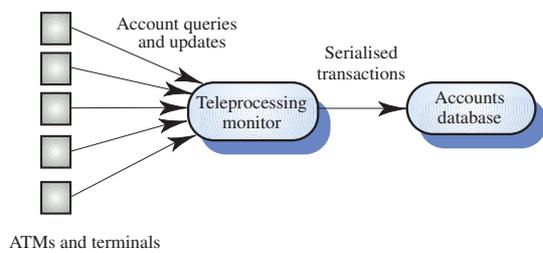
---

---

---

---

## Processamento de transações



Ofan Sommerville 2000

Engenharia de Software, 6ª edição, Capítulo 26

Slide 15

---

---

---

---

---

---

---

---

## Dados legados

- Duas questões importantes sobre legados em sistemas centrados em banco de dados
  - O sistema de gerenciamento de BD utilizado pode ser obsoleto e incompatível com os outros sistemas de gerenciamento de BD (database management systems - DBMS) utilizados pela empresa
  - O monitor de teleprocessamento utilizado pode ter sido projetado para uso com um determinado BD e mainframe. Pode não ser possível utilizar o mesmo monitor de teleprocessamento com um novo BD

---

---

---

---

---

---

---

---

## Projeto de sistemas legados

- Praticamente todos os sistemas legados utilizados foram projetados antes do desenvolvimento orientado a objetos ser utilizado
- Em vez de serem organizados com um conjunto de objetos interativos, esses sistemas são projetados utilizando uma estratégia de projeto orientado a funções
- Centenas de programas de aplicações foram desenvolvidos utilizando-se esses métodos e as ferramentas CASE associadas

---

---

---

---

---

---

---

---

## Principais sistemas de negócios

- Sistemas de processamento em lotes
  - A entrada e saída dos dados é em lotes (batches) provenientes de um arquivo, e não a partir de um terminal de usuário
- Sistemas de processamento de transações
  - As operações de entrada e saída de dados ocorrem como uma série de transações em um dado de dados, com a transação gerada a partir de um terminal de usuário

---

---

---

---

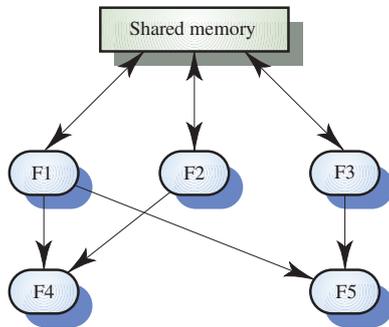
---

---

---

---

## Visão de projeto orientado a funções



©Ian Sommerville 2000

Engenharia de Software, 6ª edição, Capítulo 26

Slide 19

---

---

---

---

---

---

---

---

## Processo funcional ao projeto

- Projeto de fluxo de dados
  - Modelo de processamento no sistema utilizando DFD
- Decomposição estrutural
  - Modelo de como as funções são decompostas em sub-funções utilizando estrutura gráfica
- Projeto detalhado
  - As entidades no projeto e suas interfaces são descritas em detalhes. Estes podem ser gravados em um dicionário de dados e representados no projeto utilizando uma linguagem de modelagem.

©Ian Sommerville 2000

Engenharia de Software, 6ª edição, Capítulo 26

Slide 20

---

---

---

---

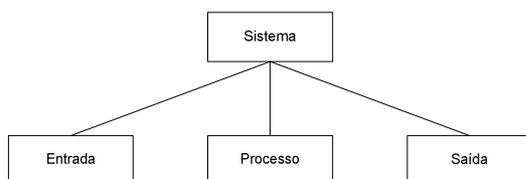
---

---

---

---

## Modelo entrada-processo-saída



©Ian Sommerville 2000

Engenharia de Software, 6ª edição, Capítulo 26

Slide 21

---

---

---

---

---

---

---

---

## Entrada-processo-saída

- Um componente de entrada pode ler alguns dados (entrada) a partir de um terminal de usuário, verificar a validade desses dados e corrigir alguns erros (processo) e pôr em fila os dados válidos, a fim de realizar o processamento (saída)
- Um componente de processamento pode iniciar uma transação a partir de uma fila (entrada), realizar alguns cálculos sobre os dados, criar um novo registro de dados anotando os resultados do cálculo (processo) e pôr na fila esse novo registro, para realizar a impressão (saída)
- Um componente de saída pode ler os registros a partir de uma fila (entrada), formatá-los de acordo com o formulário de saída (processo) e remetê-los para uma impressora (saída)
- O projeto de sistemas orientados a funções (entrada, processo e saída) pode ser modelado utilizando diagramas de fluxo de dados

---

---

---

---

---

---

---

---

## Processo funcional ao projeto

- Projeto de fluxo de dados
  - Modelo de processamento no sistema utilizando DFD
- Decomposição estrutural
  - Modelo de como as funções são decompostas em sub-funções utilizando estrutura gráfica que reflete a estrutura de entrada/processo/saída
- Projeto detalhado
  - As funções no projeto e suas interfaces são descritas em detalhes

---

---

---

---

---

---

---

---

## Diagramas de fluxo de dados

- Mostra como um item de dados de entrada é transformado funcionalmente por um sistema em um item de dados de saída
- São uma parte integrante de muitos métodos de projeto e são apoiadas por muitos sistemas CASE
- Pode ser traduzido em qualquer projeto seqüencial ou paralelo. Em um projeto seqüencial, elementos são funções ou procedimentos; em um projeto paralelo, elementos são tarefas ou processos

---

---

---

---

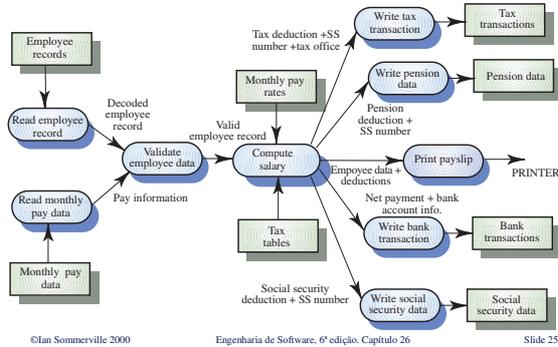
---

---

---

---

## DFD de um sistema de folha de pagamento




---

---

---

---

---

---

---

---

## Processamento de uma folha de pagamento

- As funções à esquerda do DFD são as funções de entradas
  - Leia o registro de funcionários, Leia os dados de pagamento mensal, Valide os dados do funcionário.
- A função central – Calcule o salário – realiza o processamento
- As funções à direita são as funções de saída
  - Escreva transações de impostos, Escreva dados sobre pensão, Imprima o aviso de pagamento, Escreva transação bancária, Escreva os dados de previdência social

---

---

---

---

---

---

---

---

## Processamento de transação

- Um exemplo de um sistema de processamento de transações é o software que controla os caixas automáticos de bancos (ATMs)
- Transações são independentes, pois não contam com o resultado dos serviços anteriores. Por isso, uma abordagem funcional é uma maneira natural para implementar um processamento de transação

---

---

---

---

---

---

---

---

```

INPUT
loop
  repeat
    Print_input_message (" Welcome - Please enter your card");
  until Card_input;
  Account_number := Read_card;
  Get_account_details (PIN, Account_balance, Cash_available);

PROCESS

if Invalid_card (PIN) then
  Retain_card;
  Print ("Card retained - please contact your bank");
else
  repeat
    Print_operation_select_message;
    Button := Get_button;
  case Get_button is
    when Cash_only =>
      Dispense_cash (Cash_available, Amount_dispensed);
    when Print_balance =>
      Print_customer_balance (Account_balance);
    when Statement =>
      Order_statement (Account_number);
    when Check_book =>
      Order_checkbook (Account_number);
  end case;
  Print ("Press CONTINUE for more services or STOP to finish");
  Button := Get_button;
until Button = STOP;

OUTPUT
Eject_card;
Print ("Please take your card");
Update_account_information (Account_number, Amount_dispensed);
end loop;

```

Descrição de projeto de um ATM

---

---

---

---

---

---

---

---

---

---

---

---

## Utilizando projeto orientado a funções

- Para algumas classes de sistemas, como alguns sistemas de processamento de transações, a abordagem orientada a funções pode ser melhor para projetar que uma abordagem orientada a objetos
- Empresas podem têm investido em métodos estruturados, ferramentas CASE e treinamento de pessoal para projetar orientado a funções. Os riscos e os custos de mudar para uma abordagem orientada a objetos para o projeto de programa podem não ser justificáveis

---

---

---

---

---

---

---

---

---

---

---

---

## Avaliação dos sistemas legados

- As organizações que dependem de muitos sistemas legados precisam decidir como obter o melhor retorno de seu investimento
  - Descartar completamente o sistema
  - Continuar a manter o sistema
  - Transformar o sistema de alguma maneira para melhorar sua facilidade de manutenção
  - Substituir o sistema por um novo
- A decisão deve considerar uma perspectiva de negócios (valor de negócios) e uma perspectiva de sistema (qualidade do sistema)

---

---

---

---

---

---

---

---

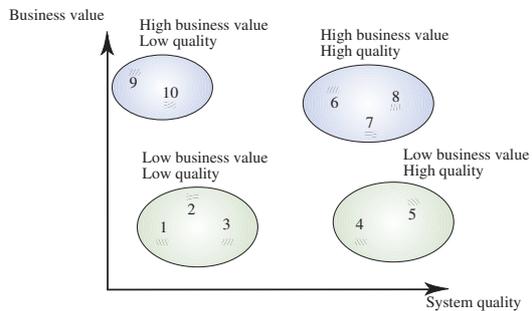
---

---

---

---

## Qualidade de sistema e valor de negócio



Ofan Sommerville 2000

Engenharia de Software, 6ª edição, Capítulo 26

Slide 31

---

---

---

---

---

---

---

---

## Grupos de sistema legados

- **Baixa qualidade, baixo valor de negócios**
  - Estes sistemas são candidatos a serem descartados
- **Baixa qualidade, alto valor de negócios**
  - Esses sistemas estão prestando uma importante contribuição à empresa, mas são dispendiosos para manter. São candidatos à transformação ou à substituição, se um sistema adequado estiver disponível
- **Alta qualidade, baixo valor de negócios**
  - Substitua por COTS, descarte completamente ou mantenha
- **Alta qualidade, alto valor de negócios**
  - Devem ser mantidos em operação, deve continuar a manutenção normal do sistema

Ofan Sommerville 2000

Engenharia de Software, 6ª edição, Capítulo 26

Slide 32

---

---

---

---

---

---

---

---

## Avaliação do valor de negócios

- A avaliação pode ter diferentes pontos de vista que devem ser considerados, como:
  - Usuários finais do sistema
  - Clientes
  - Gerentes de linha
  - Gerentes de TI
  - Gerentes seniores
- Identificados os pontos de vista, as pessoas relacionadas (stakeholders) devem ser entrevistadas e suas respostas devem ser registradas

Ofan Sommerville 2000

Engenharia de Software, 6ª edição, Capítulo 26

Slide 33

---

---

---

---

---

---

---

---

## Avaliação da qualidade do sistema

- Avaliação do processo de negócios
  - Até que ponto os processos de negócios apóiam as atuais metas da empresa?
- Avaliação do ambiente
  - Como é a eficácia do ambiente de sistema e como é dispendioso para mantê-lo?
- Avaliação do software de aplicação
  - Qual é a qualidade do software de aplicação?

---

---

---

---

---

---

---

---

## Avaliação do processo de negócio

- Para avaliar o processo, recomenda-se uma abordagem orientada a pontos de vista, comparável àquela utilizada para avaliação do valor de negócios.
- Exemplos de perguntas:
  - Existe um modelo definido do processo e existem procedimentos implantados para verificar se esse modelo é seguido?
  - As diferentes partes da organização têm os mesmos processos para as mesmas funções?
  - Como o pessoal envolvido no processo adaptou o modelo a fim de torná-lo mais adequado a seu trabalho?
  - As relações com outros processos de negócios são necessárias? Se positivo, elas são claras para o pessoal envolvido no processo?
  - O processo é compatível com o software legado de aplicação? Ele fornece as informações requeridas? O processo exige que os mesmos dados sejam fornecidos diversas vezes, em diferentes lugares?

---

---

---

---

---

---

---

---

## Avaliação do ambiente

Fator	Questões
Estabilidade do fornecedor	O fornecedor ainda existe? O fornecedor é financeiramente estável e a continuação de sua existência é provável? Se o fornecedor não existe mais, alguma outra empresa faz sua manutenção?
Taxa de falhas	O hardware tem uma alta taxa de falhas relatadas? O software de apoio 'trava' e força novas inicializações do sistema?
Idade	Qual é a idade do hardware e do software? Quanto mais antigos forem o hardware e o software de apoio, mais obsoletos eles serão. Eles podem ainda funcionar corretamente, mas pode haver uma economia significativa e também benefícios de negócios em mudar para sistemas mais modernos.
Desempenho	O desempenho do sistema é adequado? Os problemas de desempenho têm um efeito significativo nos usuários de sistema?
Requisitos de suporte	Qual suporte local é exigido pelo hardware e pelo software? Se houver altos custos associados com esse suporte, poderá valer a pena considerar a substituição do sistema.
Custos de manutenção	Quais os custos de manutenção de hardware e licenças de software de apoio? Hardwares mais antigos podem ter custos de manutenção mais altos do que os sistemas modernos. O software de apoio pode ter altos custos anuais de licença.
Interoperabilidade	Existem problemas de interface de sistema com outros sistemas? Os compiladores podem ser utilizados com versões atuais do sistema operacional? A emulação de hardware é necessária?

---

---

---

---

---

---

---

---

## Avaliação do software de aplicação

Fator	Questões
Facilidade de compreensão	Qual é o nível de dificuldade para se entender o código-fonte do sistema atual? Qual é a complexidade das estruturas de controle que são utilizadas? As variáveis têm nomes significativos que refletem sua função?
Documentação	Que documentação de sistema está disponível? A documentação está completa, consistente e atualizada?
Dados	Existe um modelo de dados explícito para o sistema? Até que ponto os dados são duplicados em arquivos diferentes? Os dados utilizados pelo sistema estão atualizados e consistentes?
Desempenho	O desempenho da aplicação é adequado? Os problemas de desempenho têm um efeito significativo nos usuários de sistema?
Linguagem de programação	Modernos compiladores estão disponíveis para a linguagem de programação utilizada para desenvolver o sistema? A linguagem de programação ainda é utilizada para novos desenvolvimentos de sistema?
Gerenciamento de configuração	Todas as versões de todas as partes do sistema são gerenciadas por um sistema de gerenciamento de configuração? Há uma descrição explícita das versões de componentes utilizados no sistema corrente?
Dados de teste	Existem dados de teste para o sistema? Há um registro de testes de regressão realizados quando novas características são acrescentadas ao sistema?
Habilidades pessoais	Há pessoas disponíveis que tenham as habilidades para a manutenção da aplicação? Existe apenas um número limitado de pessoas que compreendem o sistema?

---

---

---

---

---

---

---

---

---

---

## Sistema de medição

- É possível coletar dados quantitativos para ajudar a avaliar a qualidade do sistema de aplicação
- Exemplos de dados quantitativos que podem ser coletados:
  - O número de pedidos de modificações no sistema
  - O número de diferentes interfaces com o usuário utilizadas pelo sistema
  - O volume de dados utilizados pelo sistema

---

---

---

---

---

---

---

---

---

---

## Pontos-chave

- Um sistema legado é um sistema antigo que ainda fornece serviços de negócios essenciais
- Os sistemas legados não são apenas sistemas de aplicações, mas também incluem processos de negócios, software de apoio e hardware de sistema
- A maior parte dos sistemas legados inclui uma série de diferentes programas e dados compartilhados associados
- A maioria dos sistemas legados foi projetada a partir de uma perspectiva funcional e é composta por conjuntos de funções interativas

---

---

---

---

---

---

---

---

---

---

## Pontos-chave

---

- A maioria dos sistemas legados é constituída de sistemas de processamento em lote ou de transação. Em ambos os casos, sua organização geral pode ser representada com a utilização de um modelo entrada-processo-saída
- O valor de negócios de um sistema legado e a qualidade do software de aplicação, assim como seu ambiente, devem ser avaliados para ajudar a decidir entre substituir, transformar ou fazer a manutenção do sistema
- O valor de negócios de um sistema é uma avaliação da eficácia do sistema em fornecer apoio aos objetivos do negócio
- A qualidade do sistema depende dos processos de negócios, do software de aplicação em si e do hardware e do software utilizados para apoiar o sistema

---

---

---

---

---

---

---

---

## Bibliografia

---

- SOMMERVILLE, Ian, Engenharia de Software. 6 ed. São Paulo: Addison Wesley, 2003.
- <http://www.comp.lancs.ac.uk/computing/resources/IanS/SE6/Slides/PPT/ch26.ppt>

---

---

---

---

---

---

---

---